

فشرده سازی پایگاه داده تصاویر چهره با استفاده از الگوریتم RLS-DLA

امیر مسعود طاهری^{۱*}، همایون مهدوی نسب^{۲،†}

۱- دانشگاه آزاد اسلامی، واحد نجف آباد، گروه مهندسی برق، نجف آباد، ایران

۲- دانشگاه آزاد اسلامی، واحد نجف آباد، گروه مهندسی برق، نجف آباد، ایران

خلاصه

رشد فناوری و افزایش تصاعدی اطلاعات نیاز به حجم ذخیره سازی بیشتر را افزایش داده است. هنگامی که فشرده سازی یک دسته یا خانواده از تصاویر مدنظر باشد مانند پایگاه داده تصاویر چهره یک سازمان یا موسسه یا پایگاه داده تصاویر ام آر آی یک بیمارستان بزرگ در این صورت افزودنی اطلاعات افزایش می یابد و باعث می شود تا فشرده سازی اهمیت بیشتری پیدا کند. در این میان تصاویر چهره با توجه به کاربرد وسیعی که دارند و به عنوان رایج ترین تصاویر پرسنلی پایگاه داده سازمانهای مختلف مانند اداره پلیس، یک دانشگاه و غیره شناخته می شوند مورد توجه بیشتری قرار دارند. به همین خاطر ارائه الگوریتمی که بتواند این دسته تصاویر را با کیفیت بیشتری فشرده سازی کند اهمیت پیدا می کند. در این مقاله با استفاده از حوزه جدیدی از پردازش سیگنال ها به نام حوزه تنک و الگوریتم جدید یادگیری دیکشنری RLS-DLA روشی جدیدی برای فشرده سازی تصاویر چهره ارائه شده است که می تواند برای فشرده سازی پایگاه داده تصاویر به کار رود. نتایج بدست آمده از این الگوریتم نشان دهنده عملکرد خوب این الگوریتم و همچنین برتری آن نسبت به روش های فشرده سازی همچون JPEG و JPEG2000 می باشد.

واژه های کلیدی: فشرده سازی تصویر، کدینگ تنک، یادگیری دیکشنری، پایگاه داده تصاویر، RLS-DLA، JPEG، JPEG2000

۱. مقدمه

در سال های اخیر حوزه جدیدی به نام حوزه نمایش تنک[‡] معرفی شده است که ابزارهای موثر و مفیدی برای پردازش و آنالیز سیگنال توسط آن ارائه می شود. هدف از حوزه نمایش تنک، نمایش سیگنال ها به صورت ترکیب خطی از یک سری پایه ها و یا توابع داده شده می باشد به گونه ای که اکثر ضرایب آن برابر صفر باشد یا به عبارتی دیگر دارای نمایشی تنک باشد [3]-[1]. از خاصیت تنک بودن می توان در نمونه برداری و اندازه گیری سیگنال ها هم استفاده کرد. حسگری فشرده[§] حوزه های از پردازش سیگنال است که به این موضوع می پردازد [4]. مطابق قضیه نمونه برداری نایکوئیست-شانون می بایست تعدادی نمونه با نرخ مشخص از یک سیگنال باند محدود را در اختیار داشت تا بتوان آن را بازسازی کرد. اما اگر سیگنال مورد نظر در حوزه ای خاص تنک باشد، می توان با در اختیار داشتن تعداد بسیار کمتری

*Email: a.m.taheeri1984@gmail.com

†Email: mahdavinabasab@iaun.ac.ir

‡ Sparse Representation

§ Compressed Sensing

اندازه گیری خطی آن را بازسازی کرد. در واقع با استفاده از حسگری فشرده به جای آنکه اول با نرخ بالا نمونه برداری کرده و بعد نمونه های بدست آمده را فشرده کنند، از همان ابتدا سیگنال به نحوی "فشرده" اندازه گیری می شود. در شکل شماره ۱ یک دیاگرام نوعی از نمونه برداری به وسیله حسگری فشرده نشان داده شده است. در این دیاگرام سیگنال تنک α با ابعاد $n \times 1$ به وسیله ماتریس x با ابعاد $m \times I$ نمونه برداری می شود و سیگنال x را با اندازه ای به مراتب کوچکتر نتیجه می دهد ($m \ll n$). سیگنال اندازه گیری شده α را می توان به گیرنده ارسال کرد و اگر α به میزان کافی تنک بوده و ماتریس D با ابعاد $m \times n$ شرایط خاصی را ارضا کند، می توان α را از روی y بازسازی کرد. به الگوریتم های بازسازی α با مشاهده x و D الگوریتم های بازبازی تنک* یا کدینگ تنک[□] گفته می شود.

$$x_{m \times 1} = D_{m \times n} \alpha_{n \times 1}$$

شکل ۱- نمایش تنک سیگنال x به وسیله ماتریس D

ماتریس D در واقع در بر دارنده تمام پایه هایی است که برای نمایش داده ها استفاده می شود، این ماتریس دیکشنری[□] نامیده می شود. عموماً این ماتریس تمام فضای برداری مربوط به داده ها را پوشش می دهد. به عبارت دیگر این ماتریس برای ساخت کل یا قسمتی از داده هایی که از فضای برداری \mathbb{R}^n را پوشش می دهد. هر کدام از ستون های ماتریس دیکشنری را یک اتم[□] می نامند. در صورتی که تعداد اتم های دیکشنری برابر بعد فضای برداری باشد آنگاه آن دیکشنری را یک دیکشنری کامل** می نامند و در این صورت هر کدام از داده ها نمایشی یکتا با استفاده از اتم های دیکشنری خواهد داشت. به عنوان مثال می توان به تبدیل فوریه اشاره کرد که دارای پایه های تبدیل ثابت و متعامد هستند و هر داده دارای ضریب یکتا مخصوص به خودش خواهد بود. در صورت کامل بودن دیکشنری نمایش داده ها توسط آن یکتا خواهد بود و جواب بدست آمده لزوماً تنک نخواهد بود. اگر تعداد اتم های یک دیکشنری کامل را بیشتر کنیم، در این صورت به دیکشنری فراکامل^{††} گفته می شود. نمایش یک سیگنال با استفاده از یک دیکشنری فراکامل دارای جواب یکتا نخواهد بود. برای درک بهتر این موضوع دستگاه معادله خطی معادله ۱ را در نظر بگیرید. به دلیل اینکه ماتریس دیکشنری فراکامل است دستگاه معادله خطی در پیدا کردن α نامعین است و دارای بی شمار جواب خواهد بود. از آنجایی که به دنبال تنک ترین پاسخ ممکن برای این دستگاه هستیم. آنگاه می توان مساله را به فرم یک مساله بهینه سازی به شکل معادله ۲ نوشت، که این مساله P_0 نامیده می شود.

$$x = D\alpha \quad (1)$$

* Sparse Recovery

† Sparse Coding

‡ Dictionary

§ Atom

** Complete Dictionary

†† Overcomplete Dictionary

$$P_0: \arg \min_{\alpha} \|\alpha\|_0 \quad s.t. \quad x = D\alpha \quad (2)$$

مسئله P_0 با استفاده از الگوریتم های مختلف کدینگ تنک حل می شود که دارای انواع مختلفی مانند الگوریتم های حریصانه*، الگوریتم های محدب[†] و ... می باشند. از جمله الگوریتم های پر کاربرد کدینگ تنک می توان به الگوریتم های OMP، ORMP و LARS اشاره کرد [5],[6].

در این مقاله روشی جدید برای فشرده سازی تصاویر چهره یک پایگاه داده با استفاده از حوزه تنک و ساخت دیکشنری اختصاصی برای آن پایگاه داده معرفی شده است که می تواند به گونه ای موثر برای فشرده سازی تصاویر چهره آن پایگاه داده تصاویر به کار گرفته شود. در بخشهای بعدی در ابتدا به معرفی الگوریتم های یادگیری دیکشنری خواهیم پرداخت، در ادامه به یک الگوریتم جدید یادگیری دیکشنری می پردازیم، پس از آن به معرفی روش جدیدی برای فشرده سازی تصاویر چهره با استفاده از یادگیری دیکشنری می پردازیم سپس با استفاده از نتایج بدست آمده از شبیه سازی به بررسی و ارزیابی روش ارائه شده می پردازیم.

۲. الگوریتم های یادگیری دیکشنری

سوال اصلی که در یادگیری دیکشنری مطرح می شود این است که چگونه دیکشنری مناسب را انتخاب کنیم که در نمایش سیگنال به خوبی عمل کند؟ و پاسخ به این سوال با الگوریتم های یادگیری دیکشنری محقق می شود. الگوریتم های یادگیری دیکشنری همواره در دهه اخیر مورد توجه پژوهشگران و محققان این حوزه بوده است [9]-[7].

همانطور که در قسمت قبل گفته شد دیکشنری D ماتریسی به ابعاد $m \times n$ است که دارای n اتم به طول m هستند. در نمایش تنک سیگنال x ، می تواند به صورت ترکیب خطی یا تقریبی از اتم های دیکشنری باشد. تقریب یا بازسازی سیگنال x را با استفاده از دیکشنری D را می توان به صورت زیر نشان داد.

$$x = \hat{x} + r \quad (3)$$

$$\hat{x} = D\alpha$$

در اینجا \hat{x} برداری است که شامل ضرایبی هستند که بیشتر آنها صفر می باشند و r خطای نمایش است. هدف از یادگیری دیکشنری در واقع یافتن یک دیکشنری است که کمترین ضرایب غیر صفر را در بردار α ایجاد نماید. در یادگیری دیکشنری یکی از مهمترین مسائل دسترسی به یک مجموعه آموزش می باشد که این مجموعه شامل بردارهای آموزشی است که همگی با طول n هستند و به آنها ماتریس داده های آموزش نیز می گویند. اگر ماتریس داده های آموزش را به صورت $X = \{x_l\}_{l=1}^L, x_l \in \mathbb{R}^m$ نشان دهیم که شامل L بردار سیگنال به طول m می باشند. در این صورت هدف از یادگیری دیکشنری پیدا کردن یک ماتریس D است که توسط آن بتوان سیگنال های آموزشی را به تنک ترین حالت ممکن نمایش داد. در واقع در هنگام یادگیری دیکشنری یک مساله بهینه سازی به شکل معادله ۴ حل می شود [9].

$$\arg \min_{D, A} \|A\|_0 + \gamma \|X - DA\|_F^2 \quad (4)$$

* Greedy Algorithms

† Convex Algorithms

در اینجا X ماتریس داده های آموزشی و $A = \{\alpha_l\}_{l=1}^L, \alpha_l \in \mathbb{R}^n$ ماتریس ضرایب (وزن های) تنک متناظر با آن می باشد که در آن α_l به عنوان ستون های ماتریس ضرایب می باشد و $\|A\|_0 = \sum_{l=1}^L \|\alpha_l\|_0$ برابر مجموع تعداد ضرایب غیر صفر می باشد. همچنین $\|X - DA\|_F^2 = \sum_{l=1}^L \|r_l\|_2^2$ برابر مجموع خطاهای به وجود آمده و γ ضریب لاگرانژ می باشد. لازم به ذکر است معادله ۴ با توجه به جفت (D, A) یک مساله بهینه سازی سخت می باشد و به طور کلی در اکثر روش های یادگیری دیکشنری برای حل این مساله با استفاده از یک فرآیند بازگشتی دو مرحله ای به حل آن می پردازند. این فرآیند با یک دیکشنری اولیه آغاز می شود و به صورت زیر می باشد.

۱- با فرض ثابت بودن دیکشنری D ، ماتریس A پیدا می شود (مرحله کدینگ تنک).

$$A^{(k+1)} = \arg \min_{D, A} \|A\|_0 + \gamma \|X - D^{(k)}A\|_F^2 \quad (5)$$

۲- با فرض ثابت بودن دیکشنری A ، ماتریس D پیدا می شود (مرحله آپدیت دیکشنری).

$$D^{(k+1)} = \arg \min_{D, A} \|A^{(k+1)}\|_0 + \gamma \|X - DA^{(k+1)}\|_F^2 \quad (6)$$

در اینجا مرحله اول الگوریتم یک مساله بهینه سازی کدینگ تنک می باشد که توسط روش های کدینگ تنک حل می شود و این مرحله تقریباً در اکثر روش های یادگیری دیکشنری ثابت است. مرحله دوم تحت عنوان مرحله آپدیت دیکشنری شناخته می شود و تفاوت بین اکثر روش های یادگیری در این مرحله است که در آن خطای نمایش مرحله اول کاهش می یابد.

۳. الگوریتم RLS-DLA

الگوریتم RLS-DLA روشی جدید و کارآمد در یادگیری دیکشنری می باشد که یکی از مزیت های مهم آن این است که وابستگی به دیکشنری اولیه با استفاده از یک ضریب فراموشی λ کاهش می یابد. در این الگوریتم از یک گام زمانی i برای معرفی ماتریس داده های آموزشی استفاده می شود که به صورت $X_i = [x_1, x_2, \dots, x_i]$ با ابعاد $m \times i$ تعریف می شود. هر گام زمانی i در واقع بیان کننده یک مرحله می باشد. همانطور که مشاهده می شود در هر حلقه تکرار یکی از بردارهای آموزشی به ماتریس افزوده می شود از این رو هر زمان که بخواهیم یک بردار آموزشی جدید به ماتریس داده ها اضافه کنیم کافی است که الگوریتم را با یک مرحله یا گام اضافی اجرا کنیم و به همین خاطر این روش را یک روش یادگیری دیکشنری آنلاین^۹ می نامند. در این الگوریتم ماتریس ضرایب به صورت $A_i = [\alpha_1, \alpha_2, \dots, \alpha_i]$ با ابعاد $n \times i$ تعریف می شود. همچنین ماتریس $C_i = (A_i A_i^T)^{-1}$ تعریف می شود. در هر مرحله یا تکرار با استفاده از دیکشنری مرحله قبل یعنی D_{i-1} و یک الگوریتم کدینگ تنک مانند OMP یا ORMP وزن ها یا ضرایب متناسب با بردارهای آموزشی جدید X_i پیدا می شوند. مرحله آپدیت دیکشنری با استفاده از یک لم ماتریس معکوس بر روی ماتریس C_i و به کار گرفتن ضریب فراموشی λ به صورت زیر می باشد.

$$u = (\lambda_i^{-1} C_{i-1}) \alpha_i, \quad \lambda_i \leq 1 \quad (7)$$

* Forgetting Factor

† Time Step

‡ Online Dictionary Learning

$$\beta = \frac{1}{1 + \alpha_i^T u} \quad (8)$$

$$r_i = x_i - D_{i-1} \alpha_i \quad (9)$$

$$C_i = (\lambda_i^{-1} C_{i-1}) - \beta u u^T \quad (10)$$

$$D_i = D_{i-1} + \beta r_i u^T \quad (11)$$

لازم به ذکر است که در اینجا با به کارگیری یک ضریب فراموشی تطبیقی $\lambda_i \leq 1$ در مرحله i ام میزان وابستگی به دیکشنری اولیه کمتر می شود و باعث بهبود خواص همگرایی الگوریتم می شود. مقدار λ در صورت ثابت در نظر گرفته شدن مقداری نزدیک به 1 مثلاً حدود $1 - \frac{1}{L}$ در نظر گرفته می شود و برای λ_i تطبیقی مقدار آن به صورت بازگشتی تغییر می کند [9].

الگوریتم RLS-DLA در کاربردهایی که در آنها از داده های آموزشی بزرگ استفاده می شود، روش بسیار مناسبی است. به عنوان مثال در کاربردهای پردازش تصویر زمانی که یک تصویر به تکه های 8×8 یا 16×16 یا ... تقسیم می شود، هر تکه به صورت یک بردار آموزشی در می آید و می توان از آن به عنوان یک مجموعه آموزش استفاده کرد. برای مقدار دهی دیکشنری اولیه از n بردار اولیه آموزشی استفاده می شود و آن ها به عنوان دیکشنری اولیه در نظر گرفته می شود و توسط آن ماتریس اولیه C_i نیز بدست می آید. در این الگوریتم از معیار توقف ϵ برای تقریب خطا استفاده می شود که براساس آن می توان یک PSNR هدف* را در آغاز الگوریتم یادگیری دیکشنری محاسبه کرد. لازم به ذکر است که در این الگوریتم به کل بازگشت هایی که به تعداد بردارهای آموزش داریم یک تلاش یا آزمایش[†] گفته می شود و همچنین یک سری تعداد تکرار برای هر آزمایش مشخص می شود تا الگوریتم به PSNR مطلوب یا هدف خود همگرا شود. برای مطالعه بیشتر به [9] مراجعه شود.

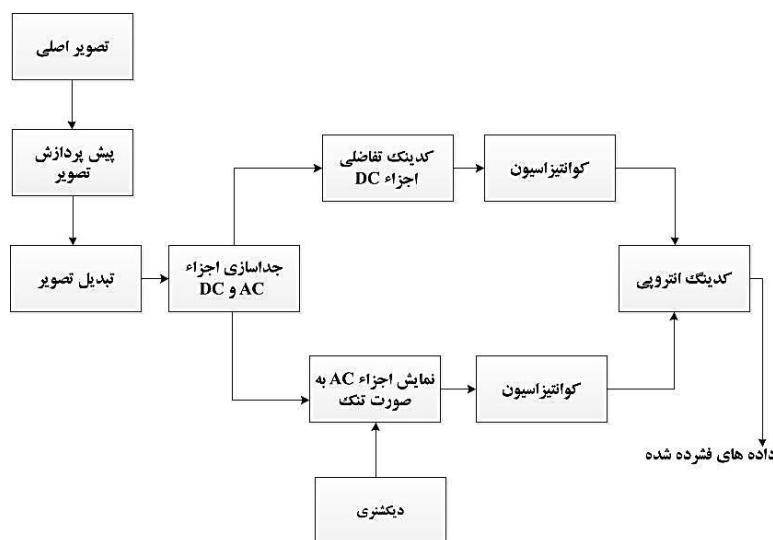
نتایج تجربی نشان می دهد که الگوریتم RLS-DLA نسبت به الگوریتم های MOD و K-SVD هم در توانایی نمایش مجموعه آموزش و در بازسازی آنها بهتر عمل می کند [8]-[10]. همچنین از مزایای دیگر آن می توان به استفاده از مجموعه های آموزش بسیار بزرگ اشاره کرد که منجر به ساخت دیکشنری هایی می شود که برای یک کلاس از سیگنال ها به کار می رود.

* Target PSNR

† Trial

۴. روش پیشنهادی

بلوک دیاگرام فرآیند فشرده سازی تصویر با استفاده از نمایش تنک و یک دیکشنری آموخته در دیکدر که در الگوریتم پیشنهادی از آن استفاده کرده ایم در شکل ۲ آمده است. این فرآیند به ترتیب شامل مراحل پیش پردازش*، تبدیل تصویر، جداسازی اجزا DC و AC، کدینگ تنک، کدینگ تفاضلی، کوانتیزاسیون و کدینگ انتروپی می باشد.



شکل ۲- بلوک دیاگرام الگوریتم فشرده سازی با استفاده از نمایش تنک

پس از فشرده شدن داده های فشرده شده به دیکدر ارسال شده و در دیکدر دقیقا عکس فرآیند صورت گرفته شده انجام می گیرد تا تصویر اصلی بازیابی شود. در ادامه به بررسی جزئیات انجام فرآیند های مختلف در روش پیشنهادی می پردازیم.

۴.۱. پیش پردازش تصویر

در بخش پیش پردازش تصویر برای بلوک بعدی آماده می شود بدین صورت که تصویر ورودی اگر تصویر رنگی باشد شامل لایه های رنگی می باشد و هر لایه رنگ جدا می شود و همچنین اگر سیاه و سفید باشد اطلاعات هر لایه به طور مستقل از دیگر لایه ها یا اطلاعات روشنایی تصویر سیاه سفید به زیر تصویرهایی[†] با اندازه 8×8 یا 16×16 یا 32×32 تقسیم می شود. در این مرحله اگر یک تصویر با ابعاد $M \times N$ پیکسل باشد اگر M یا N به ۸ یا ۱۶ یا ۳۲ تقسیم پذیر نباشند در این صورت یک یا چند ردیف با مقدار صفر به زیر تصویر یا لایه ها یا به یک یا چند ستون در سمت راست تصویر یا لایه ها اضافه می شود تا اندازه های زیر تصویرها مقدار صحیحی شوند. در بسیاری از کاربردهای فشرده سازی تصویر انتخاب ابعاد زیر تصویرها برای انجام فرآیندهای پردازش مساله حائز اهمیت است و معمولا ابعاد زیر تصویرها به صورت $2^n \times 2^n$ انتخاب می شوند. انتخاب ابعاد زیر تصویرها با ابعاد بزرگتر می تواند موجب ایجاد خطای تبدیل زیاد می شود که این مساله در بازسازی تصویر فشرده مشخص می شود. به خاطر اینکه در استانداردهای بین المللی ابعاد زیر تصویرهای

* Preprocessing

† Subimages

استفاده شده برابر 8×8 می‌باشد [11] و همچنین با استفاده از نتایج تجربی دریافتیم که بهترین نتیجه با استفاده از این ابعاد بدست می‌آید بنابراین در روش پیشنهادی ما از زیرتصویرهایی با ابعاد 8×8 استفاده می‌کنیم.

۴.۲. نحوه انجام تبدیلات تصویر

برای کاهش افزونگی تصویر می‌توان از تبدیل‌های مختلف تصویر استفاده کرد. برای انجام عمل تبدیل در فشرده‌سازی، تبدیل تصویر روی زیرتصویرها یا بلوک‌های تصویر ایجاد شده در مرحله قبل، اعمال خواهد شد. تبدیل‌های مختلفی می‌توان در این مرحله استفاده کرد که از جمله آنها می‌توان به تبدیل کسینوسی یا DCT به کار رفته در JPEG و یا تبدیلات موجک مشهور همچون تبدیل موجک $CDF 9/7$ (bi-orthogonal 4.4) یا تبدیل موجک $CDF 5/3$ (bi-orthogonal 2.2) به کار رفته در JPEG2000 اشاره کرد. به دلیل این که تبدیل DCT باعث ایجاد اثرات بلوکی می‌شود و دیگر اینکه تبدیل موجک کیفیت بهتری در فشرده‌سازی ارائه می‌کند، بنابراین تبدیل موجک $CDF 9/7$ را برای فشرده‌سازی به کار می‌بریم. با استفاده از این تبدیل موجک هر زیرتصویر را تا ۳ مرحله تجزیه می‌شود، که تعداد ۱۰ زیر باند بدست می‌آید. پس از تجزیه زیرتصویر بیشترین توزیع انرژی در ضرایب پایین گذر یا ضرایب DC می‌باشد. پس از انجام فرایند تبدیل بر روی زیرتصویر تعداد ضرایب تبدیل بدست آمده برابر با تعداد پیکسل‌های زیرتصویر می‌باشد.

۴.۳. جداسازی اجزا DC و AC

پس از بدست آمدن ضرایب تبدیلات تمامی زیرتصویرهای تصویر اصلی آنگاه اجزا DC و AC ضرایب تبدیل هر زیرتصویر مشخص می‌شوند. زیرباند $LL3$ در تبدیل موجک و همچنین اولین ضریب تبدیل DCT به عنوان ضریب DC تبدیل و بقیه به عنوان ضرایب AC در نظر گرفته می‌شوند. به عبارتی دیگر در صورت استفاده از زیرتصویرهای 8×8 یک ضریب DC و تعداد 63 ضریب AC خواهیم داشت.

۴.۴. کدینگ تفاضلی اجزاء DC

ضرایب DC مهمترین ضرایب تصاویر هستند و به تنهایی می‌توانند نمایش خام یا کلی از تصویر ارائه کنند. به دلیل اینکه پیکسل‌های مجاور در یک تصویر اغلب ضرایب مقادیر مشابهی دارند بنابراین استفاده از یک کدینگ تفاضلی مانند DPCM برای ذخیره‌سازی آنها می‌تواند مفید واقع شود به همین خاطر پس از این که اجزاء DC ضرایب تبدیل زیرتصویرهای تصویر جدا شدند، به جای اینکه مقدار هر ضریب ذخیره شود اختلاف آن با مقدار قبلی ذخیره می‌شود. لازم به ذکر است که این کدینگ یک کدینگ بی‌اتلاف است. دلیل دیگر استفاده از این کدینگ ذخیره اجزاء DC با مقادیر کوچکتر و همچنین تنک کردن یا ایجاد بیشترین صفر در رشته ضرایب DC است.

۴.۵. کدینگ تنک اجزاء AC

پس از جداسازی اجزاء AC از ماتریس ضرایب تبدیلات آنگاه این ضرایب به صورت بردارهایی ذخیره می‌شوند، پس از آن بردارهای بدست آمده با استفاده از یک کدینگ تنک و یک دیکشنری صورت تنک نمایش داده می‌شوند. برای پیدا کردن ضرایب تنک بردارهای AC از الگوریتم OMP و یا ORMP می‌توان استفاده کرد. در این مقاله به خاطر بالا بردن کیفیت سیگنال بازسازی از الگوریتم ORMP استفاده شده است. در کدینگ تنک از خطای نمایش به عنوان معیار توقف در این الگوریتم استفاده می‌شود بدین صورت که با استفاده از یک PSNR هدف داده شده، پارامتر ϵ یا معیار توقف پیدا

می‌شود. دیکشنری که برای کدینگ تنک استفاده می‌شود باید یک دیکشنری آموخته در حوزه تبدیلی باشد که در فشرده‌سازی استفاده می‌شود، به عنوان مثال در صورتی که تبدیل مورد استفاده در فشرده‌سازی تصویر، تبدیل موجک $9/7$ CDF باشد آنگاه اتم‌های دیکشنری باید ضرایب بدست آمده از تبدیلات تصاویر آموزشی در همین حوزه تبدیل یعنی تبدیل موجک $9/7$ CDF باشد. برای آموزش دیکشنری باید تصاویر آموزشی را به زیرتصویرهای مورد نظر تقسیم کرده و تبدیل مورد نظر بر روی این تصاویر اعمال شود سپس از ضرایب بدست آمده به عنوان داده‌های آموزشی برای الگوریتم یادگیری دیکشنری استفاده شود. پس از اعمال کدینگ تنک به بردارهای ضرایب AC بردارها به صورت بردارهایی تنک یعنی با کمترین عناصر غیرصفر در می‌آیند. لازم به ذکر است که هر چه بردارها تنک تر باشد عمل فشرده‌سازی تصویر به گونه‌ای بهینه‌تر انجام خواهد شد [10].

۴.۶. کوانتیزاسیون

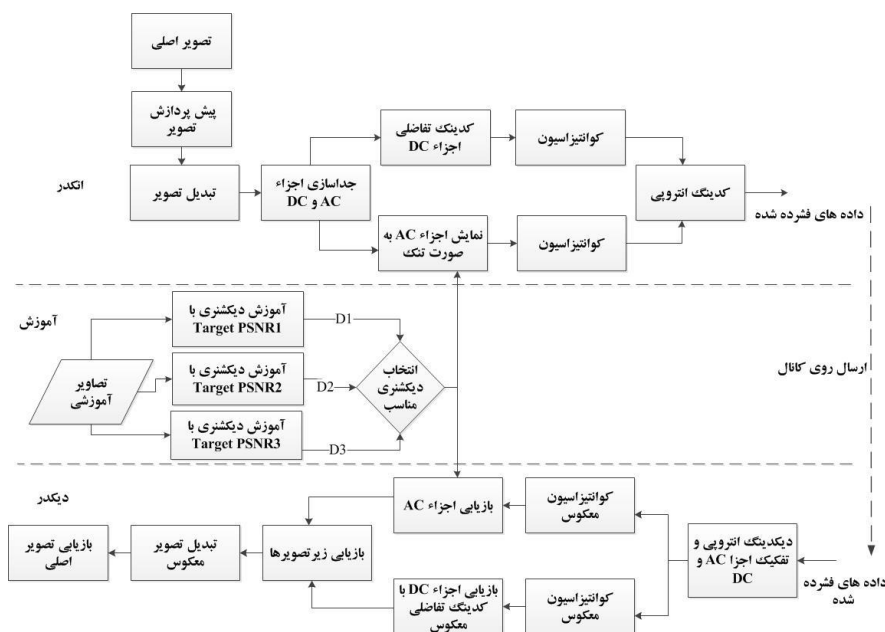
پس از اینکه ضرایب AC و DC به طور جداگانه کد شدند، کدهای بدست آمده برای هر کدام به صورت جداگانه کوانتیزه می‌شوند. برای کوانتیزاسیون آن‌ها از یک کوانتیزاسیون اسکالر یکنواخت استفاده می‌شود [11].

۴.۷. کدینگ انتروپی

ضرایب کوانتیزه شده در مرحله قبل توسط یک کدینگ انتروپی بهینه همچون کدینگ هافمن یا کدینگ حسابی به صورت بهینه کد می‌شوند سپس داده‌های فشرده نهایی بر روی کانال ارسال می‌شوند یا در یک فایل برای بازیابی مجدد ذخیره می‌شوند [12].

۴.۸. انتخاب دیکشنری مناسب

در شکل ۳ یک سیستم پیشنهادی برای بهبود عملکرد فشرده‌سازی پایگاه داده تصاویر چهره شامل انکدر و دیکدر (فشرده‌سازی و بازیابی) با استفاده از چند دیکشنری نشان داده شده است. این سیستم شامل سه بخش انکدر، آموزش و دیکدر می‌باشد. در بخش انکدر یک تصویر چهره داده شده با استفاده از یک دیکشنری انتخابی به صورتی که در قسمتهای قبل شرح داده شد، فشرده می‌شود و در بخش آموزش چند دیکشنری مختلف با PSNR هدف مختلف (معیار توقف مختلف) با استفاده از الگوریتم یادگیری دیکشنری ساخته می‌شوند سپس با استفاده از یک الگوریتم تصمیم‌گیری متناسب با کیفیت فشرده‌سازی دیکشنری مطلوب از بین آنها انتخاب می‌شود و جهت فشرده‌سازی یا بازیابی تصویر مورد نظر به کار گرفته می‌شود. در بخش دیکدر داده‌های فشرده شده به منظور بدست آوردن تصویر اولیه بازیابی می‌شوند.



شکل ۳- بلوک دیاگرام سیستم پیشنهادی برای فشرده سازی تصاویر چهره

در هنگام فشرده سازی تصاویر چهره با کیفیت های مختلف با استفاده از دیکشنری ساخته شده با الگوریتم یادگیری دیکشنری RLS-DLA به طور تجربی دریافتیم که فشرده سازی یک تصویر ، متناسب با Target PSNR دیکشنری ساخته شده در برخی بازه های کیفیت مورد نظر* عملکردی بهتر از JPEG2000 دارند و همین موضوع را مبنای کار الگوریتم تصمیم گیری قرار دادیم. رابطه کیفیت فشرده سازی مورد نظر تصویر با Target PSNR دیکشنری ساخته شده به صورت رابطه زیر می باشد:

$$Dic.TargetPSNR - 5 < Desired Compression PSNR \leq Dic.TargetPSNR \quad (12)$$

در روش پیشنهادی هنگام فشرده سازی تصویر چهره مورد نظر با استفاده از نامساوی بالا یک دیکشنری آموخته که در رابطه بالا صدق کند انتخاب می شود و آن دیکشنری برای فشرده سازی به کار گرفته می شود. مقادیر نوعی برای کیفیت تصویر فشرده شده یا مقادیر نوعی PSNR های تصویر در فشرده سازی باتلاف تصاویر چهره برای نرخ فشرده سازی 0.10 تا 2.00 بیت در پیکسل بین 35 تا 55 دسی بل می باشد، از این رو تنها ساخت سه دیکشنری برای سیستم پیشنهادی کفایت می کند. لازم به ذکر است که داده های فشرده شده در بردارنده دیکشنری های ساخته شده نیستند و فرض می شود که دیکشنری ساخته شده هم در آنکدر و هم در دیکدر موجود باشند به عبارتی دیگر در ساختار الگوریتم فشرده سازی ذخیره شوند.

* Desired Compression PSNR

۵. پیاده سازی و نتایج

برای پیاده سازی و شبیه سازی سیستم ارائه شده از نرم افزار MATLAB 2015a در یک سیستم کامپیوتر با سیستم عامل Windows 7 با پردازشگر 2 GHz و همچنین مقدار حافظه رم 8 GB استفاده شده است. برای کدینگ تنک از الگوریتم ORMP و برای یادگیری دیکشنری از الگوریتم RLS-DLA استفاده کرده ایم*. همچنین از تصاویر چهره تمام رخ موجود در پایگاه داده FEI[†] به عنوان تصاویر آموزشی و تصاویر مورد آزمایش استفاده کرده ایم. برای یادگیری دیکشنری ما از ۸ تصویر آموزشی با ابعاد 640×480 پیکسل استفاده کرده ایم. در شکل ۴ این تصاویر آموزشی نشان داده شده است. لازم به ذکر است که استفاده از تصاویر بیشتر برای آموزش دیکشنری منجر به ساخت دیکشنری بهتر خواهد شد ولی این امر باعث افزایش زمان پردازش ساخت دیکشنری می شود. در اینجا ما سه دیکشنری با استفاده از تصاویر آموزشی با Target PSNR های 55 و 50 و 40 دسی بل ساختیم که زمانی که برای پردازش هر دیکشنری طی شد به ترتیب برابر 399.1 و 315.6 و 135.7 دقیقه می باشد. هر دیکشنری شامل 440 اتم با ابعاد 8×8 پیکسل می باشد. در ساخت هر دیکشنری از هر تصویر آموزشی 3000 تکه تصویر با ابعاد 8×8 پیکسل به طور تصادفی انتخاب شده است که به طور کلی 24000 زیر تصویر برای آموزش دیکشنری ها به کار رفته است که از این زیر تصاویرها در حوزه موجک CDF 9/7 برای آموزش دیکشنری ها استفاده شده است.



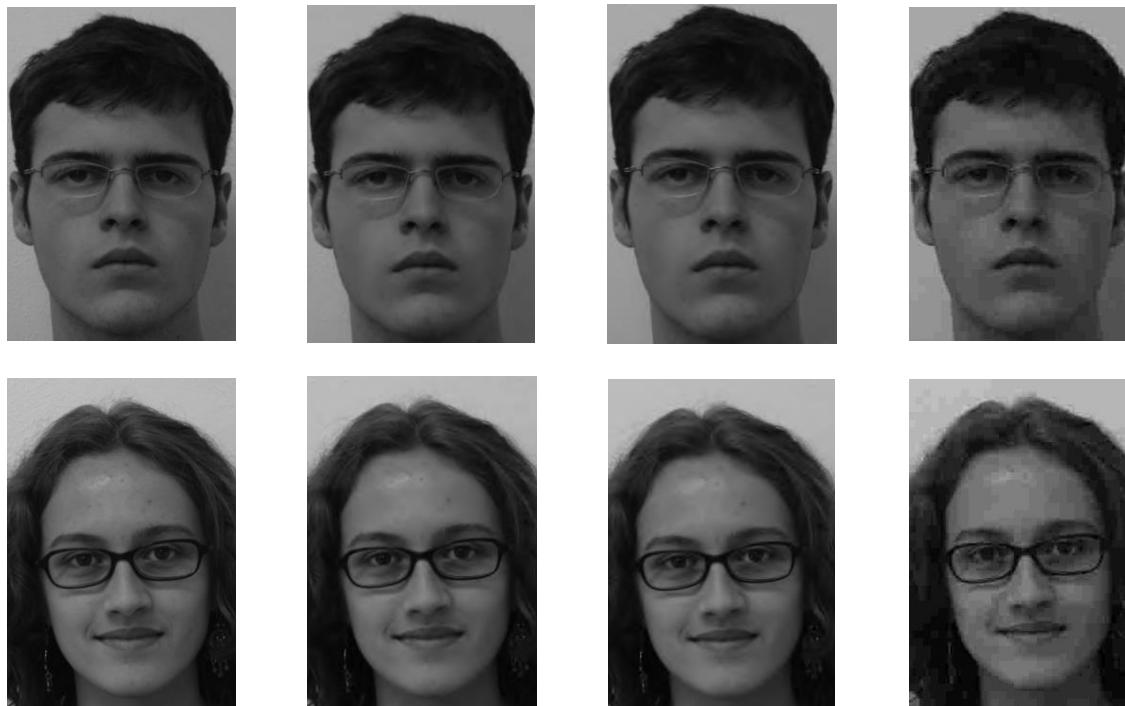
شکل ۴- مجموعه تصاویر آموزش با ابعاد 640×480 پیکسل

به منظور شبیه سازی عمل فشرده سازی از دو تصویر آزمایشی متفاوت از تصاویر مجموعه آموزش استفاده شده است. فرآیند فشرده سازی و بازیابی تصویر را بر روی این تصاویر با استفاده از الگوریتم ارائه شده و همچنین با استفاده از الگوریتم های فشرده سازی JPEG و JPEG2000 با نرخ بیت های متفاوت انجام دادیم. لازم به ذکر است که کدینگ انتروپی که در الگوریتم ارائه شده به کار بردیم کدینگ بهینه هافمن می باشد که نسبت به کدینگ انتروپی به کار رفته در JPEG2000 ساده تر و سریع تر می باشد [12]. در شکل ۵ تصاویر مورد آزمایش برای فشرده سازی و همچنین تصاویر بازیابی شده پس از فشرده سازی در نرخ بیت های یکسان با استفاده از الگوریتم های فشرده سازی مختلف نشان داده شده است و همانطور که مشاهده می شود کیفیت تصاویر بازیابی شده با استفاده از الگوریتم پیشنهادی از لحاظ بصری بهتر از دو الگوریتم دیگر می باشد. در جدول های ۱ و ۲ نتایج فشرده سازی دو تصویر آزمایشی نشان داده شده است.

* برای به کارگیری ابزارهای مختلف فشرده سازی و همچنین الگوریتم یادگیری دیکشنری از ابزارهای موجود در آدرسهای زیر استفاده کرده ایم:
<http://www.ux.uis.no/~karlsk/dle/index.html#ssec61>
<http://www.ux.uis.no/~karlsk/ICTools/ictools.html>

[†] تصاویر چهره پایگاه داده FEI از طریق آدرس زیر موجود و قابل دسترسی می باشد:

<http://fei.edu.br/~cet/face database.html>



شکل ۵- دو تصویر آزمایشی فشرده شده با نرخ 0.25 بیت در پیکسل با روش های فشرده سازی مختلف. تصاویر به ترتیب از چپ به راست: تصویر اصلی، تصویر فشرده شده با روش ارائه شده، تصویر فشرده شده با JPEG2000، تصویر فشرده شده با JPEG.

جدول ۱- نتایج کیفیت فشرده سازی (PSNR) تصویر آزمایشی ۱ با استفاده از الگوریتم های مختلف بر حسب دسی بل در نرخ بیت های مختلف

روش فشرده سازی	نرخ بیت فشرده سازی (bpp)							
	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
روش ارائه شده	44.41	46.87	48.36	49.68	50.91	52.06	53.15	54.20
JPEG2000	44.20	46.25	47.73	48.97	49.90	50.91	51.96	53.03
JPEG	41.95	44.34	46.12	47.39	48.25	48.73	49.64	51.14

جدول ۲- نتایج کیفیت فشرده سازی (PSNR) تصویر آزمایشی ۲ با استفاده از الگوریتم های مختلف بر

حسب دسی بل در نرخ بیت های مختلف

روش فشرده سازی	نرخ بیت فشرده سازی (bpp)							
	0.25	0.50	0.75	1.00	1.25	1.50	1.75	2.00
روش ارائه شده	42.30	44.72	46.53	47.95	49.14	50.48	51.58	52.67
JPEG2000	41.99	44.46	46.03	47.54	48.86	49.81	50.83	51.97
JPEG	39.47	42.40	44.08	45.72	46.35	47.33	47.91	49.00

۶. نتیجه گیری

در این مقاله یک روش جدید برای فشرده سازی پایگاه داده تصاویر چهره با استفاده از حوزه نمایش تنک و با استفاده از الگوریتم یادگیری دیکشنری RLS-DLA ارائه شد که در آن سه دیکشنری با PSNR های متفاوت از یک دسته تصاویر نمونه پایگاه داده ساخته می شود و دیکشنری مورد نظر به طور خودکار برای فشرده سازی متناسب با کیفیت فشرده سازی تصویر مورد نظر انتخاب می شود و برای فشرده سازی تصویر مورد نظر به کار می رود. در نتایج تجربی الگوریتم ارائه شده حدود 0.5 تا 1.2 دسی بل بهتر از روش هایی معروف فشرده سازی همچون JPEG2000 عمل می کند. با بکار گیری روشهای فشرده سازی انتروپی بهینه پیشرفته تر همچون کدینگ حسابی و استفاده از تصاویر آموزشی بیشتر می توان کیفیت فشرده سازی را بازم بهبود داد و در فشرده سازی یک پایگاه داده تصاویر مقدار زیادی در فضای ذخیره سازی صرفه جویی کرد. همچنین الگوریتم مورد نظر را می توان برای دیگر پایگاه های داده تصاویر با موضوع یکسان مانند پایگاه داده تصاویر اثر انگشت یا پایگاه داده تصاویر امضاء مشتریان یک بانک بکار برد.

مراجع

1. A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," SIAM, 2009.
2. Zhang, Zheng, Yong Xu, Jian Yang, Xuelong Li, and David Zhang. "A survey of sparse representation: algorithms and applications" 2015.
3. M. Elad, "Sparse and redundant representation: from theory to application in signal and image processing," Springer, 2010.
4. Donoho, David L. "Compressed sensing." Information Theory, IEEE Transactions on 52.4 (2006): 1289-1306.
5. Cotter, Shane F., et al. "Sparse solutions to linear inverse problems with multiple measurement vectors." Signal Processing, IEEE Transactions on 53.7 (2005): 2477-2488.
6. Tropp, Joel, and Stephen J. Wright. "Computational methods for sparse solution of linear inverse problems." Proceedings of the IEEE 98.6 (2010): 948-958.
7. Tošić, Ivana, and Pascal Frossard. "Dictionary learning." Signal Processing Magazine, IEEE 28.2 (2011): 27-38.
8. Engan, Kjersti, Sven Ole Aase, and J. Hakon Husoy. "Method of optimal directions for frame design." Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on. Vol. 5. IEEE, 1999.

9. Skretting, Karl, and Kjersti Engan. "Recursive least squares dictionary learning algorithm." *Signal Processing, IEEE Transactions on* 58.4 (2010): 2121-2130.
10. Skretting, Karl, and Kjersti Engan. "Image compression using learned dictionaries by RLS-DLA and compared with K-SVD." *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on.* IEEE, 2011.
11. Shi, Yun Q., and Huifang Sun. "Image and Video Compression for Multimedia Engineering: Fundamentals, Algorithms, and Standards." (2008).
12. Taubman, David, and Michael Marcellin. "JPEG2000 Image Compression Fundamentals, Standards and Practice." (2013).